AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1-29 (Cancelled)

30. (Currently Amended) A method of communicating, the method comprising:

maintaining a connection, via a network, between a first proxy on a first server and a second proxy on a second server;

while maintaining the connection:

a plurality of first processes on the first server communicating with a plurality of second processes on the second server via the connection by:

the plurality of first processes exchanging data with the first proxy via shared memory, wherein each of the plurality of first processes is assigned a unique region of the shared memory, and wherein a virtual device driver is associated with each unique region, and

wherein exchanging data with the first proxy includes, for each first process of the plurality of first processes:

the each first process writing data to the respective unique region assigned to the each first process, and the first proxy reading data from the respective unique region assigned to the each first process; and

the first proxy writing data to the respective unique region assigned to the each first process, and the each first process reading data from the

respective unique region assigned to the each first process;

wherein each first process reads from and writes to its respective unique region under regulation of the virtual device driver that is associated with the respective unique region; and wherein the first proxy reads from and writes to the respective unique regions under regulation of one or more virtual device drivers;

the first proxy exchanging the data via the connection with the second proxy; and

the second proxy exchanging the data with the plurality of second processes.

31. (Currently Amended) The method of Claim 30, wherein the plurality of first processes exchanging data with the first proxy via shared memory comprises:

a first process of the plurality of first processes writing data to a region of the shared memory that is assigned to the first process; and

the first process causing the state of a process ~~mark device~~ virtual device driver that is associated with the region that is assigned to the first process to change to a first state to indicate that the region is not writeable by the first process, wherein the ~~process mark device~~ process virtual device driver has the first state and a second state that indicates that the region is writeable by the first process.

32. (Previously Presented) The method of Claim 31, wherein the plurality of first processes exchanging data with the first proxy via shared memory further comprises:

prior to the first process writing data to the region of the shared memory that is assigned to the first process, the first process determining whether the region of the shared memory is currently writeable by the first process.

33. (Currently Amended) The method of Claim 32, wherein the first process determining whether the region of the shared memory that is assigned to the first process is currently writeable comprises the first process checking the state of the process ~~mark device~~ virtual device driver.

34. (Currently Amended) The method of Claim 33, further comprising the first process causing the state of a proxy virtual device driver ~~mark device~~ to change to a first state to indicate that the region of the shared memory that is assigned to the first process is readable by the first proxy, wherein the proxy virtual device driver ~~mark device~~ has the first state and a second state that indicates that the region that is assigned to the first process is not readable by the first proxy.

35. (Currently Amended) The method of Claim 34, wherein the first process causing the state of the proxy virtual device driver ~~mark device~~ to change to the first state comprises the first process writing to the process virtual device driver ~~mark device~~.

36. (Currently Amended) The method of Claim 34, wherein the plurality of first processes exchanging data with the first proxy via shared memory further comprises:

in response to the proxy virtual device driver ~~mark device~~ changing to the first state [[sate]], the first proxy determining that there is data to be read from the region of the shared memory that is assigned to the first process.

37. (Currently Amended) The method of Claim 34, wherein the plurality of first processes exchanging data with the first proxy via shared memory further comprises:

the first proxy reading data from the region of the shared memory that is assigned to the first process; and

the first proxy causing the proxy virtual device driver ~~mark device~~ to change to the second state.

38. (Currently Amended) The method of Claim 37, further comprising:

the process virtual device driver ~~mark device~~ changing to the second state in response to the proxy virtual device driver ~~mark device~~ changing to the second state.

39. (Currently Amended) The method of Claim 30, wherein the plurality of first processes exchanging data with the first proxy via shared memory comprises:

a first process of the plurality of first processes reading data from a region of the shared memory that is assigned to the first process; and

the first process causing the state of a process virtual device driver ~~mark device~~ to change to a first state to indicate that the region of the shared memory that is assigned to the first process is not readable by the first process, wherein the process virtual device driver ~~mark device~~ has the first state and a second state that indicates that the region of the shared memory that is assigned to the first process is readable by the first process.

40. (Currently Amended) A communication system, comprising:

a first server comprising:

a plurality of first processes;

a first proxy; and

a first shared memory having a plurality of first slots to store first data to be exchanged between the first processes and the first proxy; each first slot being assigned to a particular one of the first processes;

a plurality of process virtual device drivers, at least one virtual device driver being assigned to each first slot to regulate data flow into and out of the first slots of the shared memory;

a plurality of proxy virtual device drivers, each proxy virtual device driver corresponding to one of the process virtual device drivers;

a second server comprising:

a plurality of second processes;

a second proxy; and

a second shared memory having a plurality of second slots to store second data to be exchanged between the second processes and the second proxy; each second slot being assigned to a particular one of the second processes;

wherein the first proxy is configured to maintain a connection, via a network, with the second proxy;

wherein the first proxy and the second proxy are configured to exchange the first data and the second data via the connection to allow the plurality of first processes to communicate with the plurality of second processes; and

wherein the first proxy and the plurality of first processes exchange data by, for each first process of the plurality of first processes:

each first process writing data to a respective slot assigned to the each first process, and the first proxy reading data from the respective slot assigned to the each first process; and

the first proxy writing data to the respective slot assigned to the each first process, and the each first process reading data from the respective slot assigned to the each first process

wherein each process virtual device driver is configured to cooperate with the corresponding proxy virtual device driver to regulate data being inputted to and outputted from the corresponding first slot.

41. (Cancelled)

42. (Currently Amended) A computer readable medium having stored thereon instructions, which when executed on one or more processors, cause the one or more processors to perform the steps of:

maintaining a connection, via a network, between a first proxy on a first server and a second proxy on a second server;

while maintaining the connection:

a plurality of first processes on the first server communicating with a corresponding plurality of second processes on the second server via the connection by:

the plurality of first processes exchanging data with the first proxy via shared memory, wherein each of the plurality of first processes is assigned a unique region of the shared memory, and wherein a virtual device driver is associated with each unique region, and

wherein exchanging data with the first proxy includes, for each first process of the plurality of first processes:

the each first process writing data to the respective unique region assigned to the each first process,

and the first proxy reading data from the respective unique region assigned to the each first process; and

the first proxy writing data to the respective unique region assigned to the each first process, and the each first process reading data from the respective unique region assigned to the each first process; ~~and~~

wherein each first process reads from and writes to its respective unique region under regulation of the virtual device driver that is associated with the respective unique region; and wherein the first proxy reads from and writes to the respective unique regions under regulation of one or more virtual device drivers;

the first proxy exchanging the data via the connection with the second proxy; and

the second proxy exchanging data with the plurality of second processes.

43. (Currently Amended) The computer readable medium of Claim 42, wherein the step of the plurality of first processes exchanging data with the first proxy via shared memory comprises:

a first process of the plurality of first processes writing data to a region of the shared memory that is assigned to the first process; and

the first process causing the state of a process ~~mark device~~ virtual device driver that is associated with the region that is assigned to the first process to change to a first state to indicate that the region is not writeable by the first process,

wherein the ~~process mark device~~ process virtual device driver has the first state and a second state that indicates that the region is writeable by the first process.

44. (Previously Presented) The computer readable medium of Claim 43, wherein the step of the plurality of first processes exchanging data with the first proxy via shared memory further comprises:

prior to the first process writing data to the region of the shared memory that is assigned to the first process, the first process determining whether the region of the shared memory is currently writeable by the first process.

45. (Currently Amended) The computer readable medium of Claim 44, wherein the step of the first process determining whether the region of the shared memory that is assigned to the first process is currently writeable comprises the first process checking the state of the ~~process mark device~~ process virtual device driver.

46. (Currently Amended) The computer readable medium of Claim 45, wherein the method further comprises the step of the first process causing the state of a proxy virtual device driver ~~mark device~~ to change to a first state to indicate that the region of the shared memory that is assigned to the first process is readable by the first proxy, wherein the proxy mark device has the first state and a second state that indicates that the region of the shared memory that is assigned to the first process is not readable by the first proxy.

47. (Currently Amended) The computer readable medium of Claim 46, wherein the step of the first process causing the state of the proxy virtual device driver ~~mark device~~ to change

to the first state comprises the first process writing to the process virtual device driver ~~mark device~~.

48. (Currently Amended) The computer readable medium of Claim 46, wherein the step of the plurality of first processes exchanging data with the first proxy via shared memory further comprises:

in response to the proxy virtual device driver ~~mark device~~ changing to the first sate, the first proxy determining that there is data to be read from the region of the shared memory that is assigned to the first process.

49. (Currently Amended) The computer readable medium of Claim 46, wherein the step of the plurality of first processes exchanging data with the first proxy via shared memory further comprises:

the first proxy reading data from the region of the shared memory that is assigned to the first process; and

the first proxy causing the proxy virtual device driver ~~mark device~~ to change to the second state.

50. (Currently Amended) The computer readable medium of Claim 49, wherein the method further comprises the step of:

the process virtual device driver ~~mark device~~ changing to the second state in response to the proxy virtual device driver ~~mark device~~ changing to the second state.

51. (Currently Amended) The computer readable medium of Claim 42, wherein the step of the plurality of first processes exchanging data with the first proxy via shared memory comprises:

a first process of the plurality of first processes reading data from a region of the shared memory that is assigned to the first process of the shared memory that is assigned to the first process; and

the first process causing the state of a process virtual device driver ~~mark device~~ to change to a first state to indicate that the region is not readable by the first process, wherein the process virtual device driver ~~mark device~~ has the first state and a second state that indicates that the region of the shared memory that is assigned to the first process is readable by the first process.

52. (Previously Presented) The method of Claim 30 wherein the second proxy exchanging data with the plurality of second processes includes the second proxy exchanging data with the plurality of second processes via shared memory.